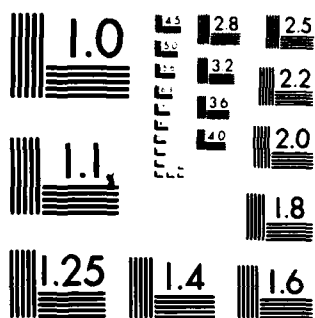


IMPLEMENTATION OF A SUBGRADIENT PROJECTION ALGORITHM II 1/1  
(U) WISCONSIN UNIV-MADISON MATHEMATICS RESEARCH CENTER  
R W OWENS AUG 83 MRC-TSR-2550 DAAG29-80-C-0041

F/G 12/1 NL

END  
DATE  
FILMED  
10 83  
DTIC



MICROCOPY RESOLUTION TEST CHART  
NATIONAL BUREAU OF STANDARDS-1963-A

2

AD-A 132 835-

MRC Technical Summary Report #2550

IMPLEMENTATION OF A SUBGRADIENT  
PROJECTION ALGORITHM II

Robert W. Owens

**Mathematics Research Center  
University of Wisconsin—Madison  
610 Walnut Street  
Madison, Wisconsin 53705**

August 1983

(Received July 12, 1983)

**DTIC FILE COPY**

Sponsored by

U. S. Army Research Office  
P. O. Box 12211  
Research Triangle Park  
North Carolina 27709

Approved for public release  
Distribution unlimited

*Signature*  
**DTIC**  
**ELECTE**  
SEP 23 1983  
**D**

83 09 22 148

UNIVERSITY OF WISCONSIN-MADISON  
MATHEMATICS RESEARCH CENTER

IMPLEMENTATION OF A SUBGRADIENT  
PROJECTION ALGORITHM II

Robert W. Owens\*

Technical Summary Report #2550  
August 1983

ABSTRACT

↓  
This paper discusses the implementation of a subgradient projection algorithm due to Sreedharan [13] for the minimization, subject to a finite number of smooth, convex constraints, of an objective function which is the sum of a smooth, strictly convex function and a piecewise smooth convex function. Computational experience with the algorithm on several test problems and comparison of this experience with previously published results is presented.  
↖

AMS (MOS) Subject Classification: 90C25, 65K05

Key Words: optimization, subgradient, convex function, algorithm

Work Unit Number 5 - Optimization and Large Scale Systems

C.R. Category: G.1.6

---

\* Department of Mathematics, Lewis and Clark College, Portland, OR 97219 USA.

---

Sponsored by the United States Army under Contract No. DAAG29-80-C-0041.

## SIGNIFICANCE AND EXPLANATION

The need to optimize an objective function subject to some sort of constraints arises in almost all fields that use mathematics. In many problems, however, the objective function is not differentiable so that standard algorithms based upon gradients are not applicable. This is the case, for example, with many problems arising in economics and business, and with certain design problems in engineering. One approach to addressing this difficulty is to construct a suitable generalized gradient, the subgradient, and base an algorithm upon it instead of on the gradient.

This paper discusses the implementation of a subgradient projection algorithm for the minimization of a certain kind of non-differentiable, nonlinear, convex function subject to a finite number of nonlinear, convex constraints. Computational experience with the algorithm on several test problems and comparison of this experience with previously published results is presented.

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By _____	
Distribution/ _____	
Availability Codes	
Dist	Avail and/or Special
A	



---

The responsibility for the wording and views expressed in this descriptive summary lies with MRC, and not with the author of this report.

# IMPLEMENTATION OF A SUBGRADIENT PROJECTION ALGORITHM II

Robert W. Owens\*

## 1. Introduction

In the early 1960's, Rosen developed gradient projection algorithms for nonlinear programming problems subject to linear [9] and nonlinear [10] constraints. Recent work by Sreedharan has generalized this earlier work and may be viewed as the subgradient counterpart of Rosen's work. In [12], Sreedharan developed a subgradient projection algorithm for the minimization of the sum of a piecewise-affine convex function and a smooth, strictly convex function, subject to a finite number of linear constraints. Rubin [11] reported on the implementation of that algorithm. Then in [13], Sreedharan extended his previous results to the case of an objective function which is the sum of a piecewise smooth convex function and a smooth, strictly convex function, subject to a finite number of smooth, convex constraints. Parallel to Rubin's investigations, this paper reports on the implementation of that algorithm. Since this latter paper by Sreedharan is an extension and generalization of the former, the computational results of Rubin are relevant to this work; consequently, where reasonable, we have chosen to parallel Rubin's presentation for ease of comparison of the results.

Considerable effort has been and continues to be expended minimizing convex functions subject to convex constraints and the difficulties encountered are numerous. While no implementable algorithm can successfully

---

\* Department of Mathematics, Lewis and Clark College, Portland, OR 97219 USA.

handle all problems, Sreedharan's algorithm does address several which have plagued others in the field. Rosen's original iterative algorithms were susceptible to what is sometimes called jamming or zigzagging, i.e. the generated sequence clustering at or converging to a nonoptimal point. Various techniques have been proposed to avoid this, by using  $\epsilon$ -subgradients. Sreedharan's algorithm is guaranteed convergence to the optimal solution. Others who have used  $\epsilon$ -subgradient projection methods, e.g. [3], encounter computational difficulties since the complete  $\epsilon$ -subdifferential uses non local information and its actual determination can be computationally prohibitive. Sreedharan's algorithm requires that only a certain subset of the  $\epsilon$ -subdifferential be computed, a more straightforward task. Polak's [7] gradient projection algorithm essentially projects the gradient of the objective function onto supporting tangent vector spaces, a method different than Sreedharan's. Moreover, Polak hypothesizes a certain linear independence of the gradients of the  $\epsilon$ -binding constraints whereas the algorithm we are reporting on requires no such assumption.

## 2. Problem

Let  $\Omega \subset \mathbb{R}^d$  be a nonempty, open, convex set, and let  $f, g_i, v_j$ ,  $i=1,2,\dots,m, j=1,2,\dots,n$  be convex differentiable functions from  $\Omega$  into  $\mathbb{R}$ . Also, let  $f$  be strictly convex. Let  $X = \{x \in \Omega \mid g_i(x) < 0, i=1,2,\dots,m\}$ . We assume that  $X$  is bounded and satisfies Slater's constraint qualification: there exists  $a \in X$  such that  $g_i(a) < 0, i=1,2,\dots,m$ . Let  $v(x) = \max\{v_j(x) \mid j=1,2,\dots,n\}$ . The problem to be solved, referred to as problem (P) is:

$$\begin{array}{ll} \text{minimize:} & f(x) + v(x) \\ \text{(P):} & \\ & \text{subject to: } x \in X \end{array}$$

Note that (P) has a unique solution since  $f + v$  is continuous and strictly convex on the compact set  $X$ . Note also, however, that in general  $f + v$  is not differentiable.



### 3. Notation

Given  $x, y \in \mathbb{R}^d$ , we denote the standard Euclidean inner product of  $x$  and  $y$  by juxtaposition, i.e.  $xy$ , and the corresponding length of  $x$  is denoted by  $|x|$ . Given a nonempty, closed convex set  $S \subset \mathbb{R}^d$ , we denote by  $N[S]$  the unique point of  $S$  of least norm.  $y = N[S]$  is the projection of  $0$  onto  $S$  and is characterized by the property that  $y(x-y) > 0$  for all  $x \in S$ . For a differentiable function  $\phi$ , denote by  $\phi'(x)$  the gradient of  $\phi$  at  $x$ . Given any set  $S \subset \mathbb{R}^d$ , let cone  $S$  be the convex cone with apex  $0$  generated by  $S$  and  $\text{conv } S$  the convex hull of the set  $S$ .

For any point  $x \in X$  and any  $\epsilon > 0$ , define the following four sets:

$$I_\epsilon(x) = \{i | g_i(x) > -\epsilon, \quad 1 \leq i \leq m\}$$

$$J_\epsilon(x) = \{j | v_j(x) > v(x) - \epsilon, \quad 1 \leq j \leq n\}$$

$$C_\epsilon(x) = \text{cone } \{g'_i(x) | i \in I_\epsilon(x)\}$$

$$K_\epsilon(x) = \text{conv } \{v'_j(x) | j \in J_\epsilon(x)\}.$$

With  $x \in X$ ,  $I_0(x)$  is the set of active (i.e. binding) constraints at  $x$ , while  $I_\epsilon(x)$ , with  $\epsilon > 0$ , is the set of  $\epsilon$ -active (i.e. almost binding) constraints at  $x$ .  $J_\epsilon(x)$ , with  $\epsilon > 0$  and  $x \in X$ , is similarly interpreted in terms of the maximizing functions that define the function  $v$ .  $C_\epsilon(x)$  and  $K_\epsilon(x)$  are the easily computable subsets of the  $\epsilon$ -subdifferentials referred to in section 1.

#### 4. Algorithm

Step 1: Carry out an unconstrained minimization of  $f$ . If no minimizer exists in  $\Omega$ , go to step 3; if the minimizer  $c$  exists in  $\Omega$  but  $c \notin X$ , go to step 3. If  $c \in X$ , proceed to step 2.

Step 2: If  $v'_j(c) = 0$  for  $j=1,2,\dots,n$ , then STOP;  $c$  solves (P). Otherwise proceed to step 3.

Step 3: Start with an arbitrary  $x_0 \in X$ ,  $\delta > 1$ , and set  $k = 0$ . Let  $0 < \epsilon_0 < -\max\{g_i(a) \mid 1 \leq i \leq m\}$ . Set  $\epsilon = \epsilon_0$ .

Step 4: Compute  $y_0 = N[f'(x_k) + K_0(x_k) + C_0(x_k)]$ . If  $y_0 = 0$ , STOP;  $x_k$  solves (P). Otherwise proceed to step 5.

Step 5: Compute  $y_\epsilon = N[f'(x_k) + K_\epsilon(x_k) + C_\epsilon(x_k)]$ .

Step 6: If  $\|y_\epsilon\|^2 > \epsilon$ , then set  $\epsilon_k = \epsilon$ ,  $S_k = y_\epsilon$ , and go to Step 8.

Step 7: Replace  $\epsilon$  by  $\epsilon/\delta$  and go to step 5.

Step 8: Let  $I = I_{\epsilon_k}(x_k)$ . If  $I = \emptyset$ , then let  $u_k = 0$  and  $M_k = 0$ . If  $I \neq \emptyset$ , then proceed as follows. Let  $\gamma_{ij} = g'_i(x_k)g'_j(x_k)$  for  $i, j \in I$  and solve the following linear programming problem for  $\bar{\mu}_i > 0$ ,  $i \in I$ :

$$\text{Minimize: } \sum_{i \in I} \mu_i$$

$$\text{Subject to: } \sum_{i \in I} \gamma_{ij} \mu_i > \|g'_j(x_k)\|, \quad j \in I.$$

$$\mu_i > 0, \quad i \in I$$

$$\text{Set } u_k = \sum_{i \in I} \bar{\mu}_i g'_i(x_k) \text{ and } M_k = (\|f'(x_k)\| + \max_{1 \leq j \leq n} \|v'_j(x_k)\|) \|u_k\|.$$

$$\text{Step 9: Let } \lambda_k = \|S_k\|^2 / (2M_k + 1) \text{ and } t_k = S_k + \lambda_k u_k.$$

$$\text{Step 10: Find } \bar{\alpha}_k = \max\{\alpha \mid x_k - \alpha t_k \in X \text{ and } g_i(x_k - \alpha t_k) < g_i(x_k) \forall i \in I\}.$$

Step 11: Find  $\alpha_k \in [0, \bar{\alpha}_k]$  such that there exists

$z_k \in f'(x_k - \alpha_k t_k) + K_0(x_k - \alpha_k t_k)$  with  $z_k t_k = 0$ . If no such  $\alpha_k$  exists, then set  $\alpha_k = \bar{\alpha}_k$ .

Step 12: Set  $x_{k+1} = x_k - \alpha_k t_k$ , increment  $k$  by 1, and go to step 4.

Steps 1 and 2 are present for technical reasons and rule out the possibility that a given problem has an easily obtained solution. For many of the test problems considered, these steps were bypassed since a specific starting point was desired or it was known that a trivial solution did not exist. In the fully implemented version of this algorithm actually tested, step 1 was performed by calling the IMSL subroutine ZXCGR which employs a conjugate gradient method to find the unconstrained minimum of an  $n$  variable function of class  $C^1$ .

The stopping criteria of either step 2 or step 4 would, in practice, be replaced by  $\|y_0\|$  or  $\|v_j(c)\|$  becoming sufficiently small.

In [13] it is shown that  $-t_k$ , from step 9, is a feasible direction of strict descent for  $f + v$  and that  $\alpha_k$ , from step 11, is strictly positive.

Before describing computational trials on test problems, the more substantial subproblems involved in carrying out the steps of the given algorithm are discussed. Specifically, we mention the projection problem inherent in both steps 4 and 5, the linear programming problem of step 8, the line search required in step 10, and the one dimensional minimization indirectly called for in step 11.

## 5. Subproblems

The problem of computing the projection of 0 onto  $f'(x_k) + K_\epsilon(x_k) + C_\epsilon(x_k)$  with  $\epsilon = 0$  (step 4) or  $\epsilon > 0$  (step 5) is solved almost exactly as in [11], i.e. the problem is expressed as a quadratic programming problem which is then solved by standard techniques. For completeness we outline the key ideas; for details see Rubin [11] and Cottle [4].

Assume that  $x \in X$ ,  $\epsilon > 0$ ,  $I_\epsilon(x) = \{i_1, \dots, i_p\} \neq \emptyset$ , and  $J_\epsilon(x) = \{j_1, \dots, j_q\} \neq \emptyset$ .  $y = N[f'(x) + K_\epsilon(x) + C_\epsilon(x)]$  can be written as  $y = f'(x) + \sum_{k=1}^p \alpha_k g'_{i_k}(x) + \sum_{k=1}^q \beta_k v'_{j_k}(x)$ , where  $\alpha_k > 0$ ,  $k=1, \dots, p$ ,  $\beta_k > 0$ ,  $k=1, \dots, q$ ,  $\sum_{k=1}^q \beta_k = 1$ , and  $(1/2) \|f'(x) + \sum_{k=1}^p \alpha_k g'_{i_k}(x) + \sum_{k=1}^q \beta_k v'_{j_k}(x)\|^2$  is a minimum over all such  $\alpha_k$  and  $\beta_k$ , s. Let  $h_k = f'(x) + v'_{j_k}(x)$ ,  $k=1, \dots, q$ . Then  $f'(x) + \sum_{k=1}^q \beta_k v'_{j_k}(x) = \sum_{k=1}^q \beta_k h_k$ . Let  $e = (0, \dots, 0, 1, \dots, 1)$  be the  $p+q$  vector the first  $p$  of whose components are 0 and the last  $q$  of whose components are 1. Let  $u = (\alpha_1, \dots, \alpha_p, \beta_1, \dots, \beta_q)$ ,  $M = (g'_{i_1}(x), \dots, g'_{i_p}(x), h_1, \dots, h_q)$ , and  $Q = M^T M$ , where the superscript  $T$  denotes the transpose. The projection problem under consideration can now be expressed as the quadratic programming problem:

Minimize:  $1/2 u Q u$

Subject to:  $ue = 1$

$u > 0$

Rubin adapted an algorithm of Wolfe to solve this problem, an algorithm which is guaranteed to terminate at the solution in finitely many steps. In the present work, a package [9] developed at the University of Wisconsin-Madison, based on Cottle's version of the principal pivoting method [4] was used.

Since the primary means of reporting on the success of the subgradient projection algorithm is to give the number of iterations needed to obtain a fixed accuracy, little extraordinary effort was expended optimizing the computations involved in each iteration. Instead, IMSL subprograms were used where convenient.

Step 8 requires the solution of the linear programming problem stated in section 4; this problem is the dual of a standard linear programming problem for which subroutine ZX3LP, a "revised simplex algorithm - easy to use version," is immediately applicable. No difficulties were encountered.

Step 10 involves a line search to locate  $\bar{\alpha}_k$ , the maximum  $\alpha$  such that  $x_k - \alpha t_k \in X$  and no previously  $\epsilon$ -binding constraint becomes more binding. Letting

$$G_i(\alpha) = \begin{cases} g_i(x_k - \alpha t_k) & i \notin I \\ g_i(x_k - \alpha t_k) - g_i(x_k) & i \in I \end{cases}$$

and  $G(\alpha) = \max\{G_i(\alpha) | 1 \leq i \leq m\}$ , and using the convexity of the  $g_i$ , step 10 can be rephrased as: find the unique positive root  $\bar{\alpha}_k$  of  $G(\alpha) = 0$ . After locating  $0 < \alpha_1 < \alpha_2$  such that  $G(\alpha_1) < 0 < G(\alpha_2)$ , this latter problem is solved by bisection with the stopping criteria being that  $0 < G(\alpha) < 10^{-12}$  and  $|\alpha_1 - \alpha_2| < 10^{-12}$ .

Step 11 requires that we locate, if it exists,  $\alpha_k \in [0, \bar{\alpha}_k]$  such that there exists  $z_k \in f'(x_k - \alpha_k t_k) + K_0(x_k - \alpha_k t_k)$  with  $z_k t_k = 0$ ; if no such  $\alpha_k$  exists, then set  $\alpha_k = \bar{\alpha}_k$ . This is equivalent to locating the minimum of the strictly convex function  $F(\alpha) = f(x_k - \alpha t_k) + v(x_k - \alpha t_k)$  on  $[0, \bar{\alpha}_k]$ ; see [13, Lemma 5.22]. The IMSL subroutine ZXGSP for one-dimensional unimodal

function minimization using the Golden Section search method was used to solve this subproblem. The stopping criterion employed was that the interval within which the minimum occurs should be less than the minimum of  $10^{-12}$  and  $10^{-4} \cdot \bar{\alpha}_k$ . Occasionally an error message appeared saying that the function did not appear unimodal to the subroutine; this was clearly due to roundoff errors becoming significant. Nevertheless, these warnings posed no major difficulties.

With the earlier, linear version of Sreedharan's algorithm, Rubin [11] was able to carry out an efficient search for  $\alpha_k$  without explicitly locating  $\bar{\alpha}_k$  by exploiting the linearity of the  $g_i$  and  $v_j$  functions. Although some improvement for finding  $\alpha_k$  in the present implementation is undoubtedly possible, no such dramatic economies are to be expected due to the nonlinearity of the functions involved.

## 6. Computational Trials

The subgradient projection algorithm was implemented in Fortran 77 on a VAX 11-780 using double precision (12-13 digits accuracy) and employing the software described in section 5. For test purposes,  $\delta$  (step 3) was taken to be 10. In order to avoid taking unnecessarily small steps early in a problem,  $\epsilon$  was reset to  $\epsilon_0$  at step 4 for the first 4 iterations and was allowed to follow the given nonincreasing pattern thereafter. The convergence criterion of step 4 was relaxed to produce termination if either  $\|y_0\| < 10^{-12}$  or  $\|x_k - x_{k+1}\| < (1/2) \cdot 10^{-9}$ .

The first two problems below are ones considered by Rubin [11], neither of which meet the hypothesis that  $f$  be strictly convex since in each  $f \equiv 0$ .

Example 1. The first example was proposed by Wolfe [14] and uses  $d = 2$ ,  $f \equiv 0$ ,  $v_1(x,y) = -x$ ,  $v_2(x,y) = x + y$ ,  $v_3(x,y) = x - 2y$ ,  $n = 3$ . Following Rubin, we investigated cases with constraints leading to a rectangular feasible region having the global minimizer  $(0,0)$  of  $f + v$  interior to, on the boundary of, and exterior to the feasible region. Table I summarizes the results. The lower left and upper right corners of the feasible region are given, as is the point at which the optimum value is achieved and the number of iterations required to obtain termination. In each case

$$\epsilon_0 = -1/2 \max_{1 \leq i \leq 4} g_i(x_0, y_0).$$

The slow convergence occurs only, but not always, in cases where the optimal solution is at a corner. The linear version of this algorithm was able to handle all cases in at most 2 iterations [11].

Lower Left Corner	Upper Right Corner	Solution	Number of Iterations
(-1,-1)	( 1, 1)	( 0, 0)	0
( 1,-1)	( 2, 1)	( 1, 0)	1
(-3, 1)	( 2, 3)	(-1/2,1)	2
(-2,-1)	(-1, 1)	(-1, 0)	1
(-2,-4)	( 2,-2)	(-2,-2)	18
( 1, 2)	( 3, 4)	( 1, 2)	13
(-2, 1)	(-1, 2)	(-1,3/2)	1
(-4,-3)	(-1,-1)	(-1,-1)	17
( 0, 0)	( 1, 1)	( 0, 0)	1
( 0,-1)	( 1, 1)	( 0, 0)	1

Table I

Test results for example 1.

Example 2. The second example, attributed to Powell, also appears in [14].

In that paper, an unconstrained minimization algorithm proposed by Wolfe achieves only linear convergence when applied to this example.  $d = 2$ ,

$f \equiv 0$ ,  $n = 5$  with  $v_i(x,y) = (\cos \frac{2\pi i}{5})x + (\sin \frac{2\pi i}{5})y$  for  $i=1,\dots,5$ . We set  $X = \text{conv}\{(\pm 1, \pm 1)\}$ , so that the global minimizer  $(0,0)$  is interior to  $X$ , and we choose the initial point  $(\rho \cos \frac{\pi}{5}, \rho \sin \frac{\pi}{5})$  for  $\rho > 0$  as was considered by both Wolfe and Rubin. Computational results for various values and ranges of  $\rho$  are summarized in table II.  $\rho < \sec \frac{\pi}{5} = 1.23605$  yields an initial point inside of  $X$  while  $\rho > \sec \frac{\pi}{5}$  puts  $x_0$  outside of  $X$ .



Rubin also tested the same function  $f + v$  with the feasible region determined by  $x, y \leq 0$  and  $x + y \leq -1$  and found convergence in 2 to 4 iterations from various starting points. Our results are in basic agreement. We had to start quite far out in the third quadrant, e.g.  $(-10, -10)$  before the algorithm required 5 iterations to terminate.

$\rho$	No. of Iterations
$[0, \sec \frac{\pi}{5})$	1
$(\sec \frac{\pi}{5}, 32.94)$	2
$(32.95, 100]^*$	3
1234	10

Table II

Test results for example 2.

\* 100 was the largest value tested in this range.

Example 3. A third example tested comes from [2, p.76] where the problem is solved by a parametric feasible direction (PFD) algorithm - section 7 of the book. The problem involves a quadratic objective function in 5 variables and 7 constraints involving linear, quadratic, and exponential terms;  $v \equiv 0$  since their algorithm is not designed to handle nondifferentiable functions. Our results are in qualitative agreement. The PFD algorithm terminated after

35 iterations with 6 decimal figures of accuracy; the subgradient projection algorithm terminated after 41 iterations with 9 decimal figures of accuracy.

Example 4. The fourth example tested was introduced by Demyanov and Malozemov [5] to illustrate jamming of a certain subgradient optimization algorithm. In this example,  $d = 2$ ,  $f(x,y) \equiv 1$  (not strictly convex),  $n = 3$  with  $v_1(x,y) = -5x + y$ ,  $v_2(x,y) = x^2 + y^2 + 4y$ ,  $v_3(x,y) = 5x + y$ . We also impose the artificial constraint  $g(x,y) = x^2 + y^2 - 99 < 0$ . Starting from any initial point on the circle  $(x + 5/2)^2 + (y + 3/2)^2 = \frac{17}{2}$ , with  $-1.928 < x < -1.831$  and  $y > 0$ , at which  $v_1$  and  $v_2$  are maximands for  $v$ , Demyanov and Malozemov note that the usual subgradient optimization algorithm converges to the nonoptimal point  $(0,0)$ , not to  $(0,-3)$  as it should.

When our subgradient projection algorithm was used to solve the above problem with nine different starting point satisfying the above conditions, the optimum solution was obtained in at most 4 iterations. For various other starting points, each of which had  $v_1$  and  $v_2$  maximands for  $v$ , the algorithm required from 1 to 7 iterations before termination. In all cases,  $\epsilon_0 = 10^{-2}$  was used.

Example 5. The fifth example, appearing in [1, p.447], has no optimum solution, but a straightforward application of a steepest feasible descent direction method without "anti-zigzagging" precautions leads to convergence. The problem is

$$\begin{aligned} \text{Minimize:} \quad & f(x,y,z) = \frac{4}{3} (x^2 - xy + y^2)^{3/4} - z \\ \text{Subject to:} \quad & x,y,z > 0 \end{aligned}$$

and with the initial point specified to be  $(0, 1/4, 1/2)$ . Clearly  $f(0,0,z)$ ,  $z > 0$ , is negative and arbitrarily large in magnitude for appropriately

chosen  $z$ . It is easily shown that, when the steepest feasible descent direction method is employed, the iterates  $(x_k, y_k, z_k)$  converge to  $(0, 0, \frac{4+\sqrt{2}}{4})$  with the third coordinate monotonically increasing. For test purposes, we introduced the extra, artificial constraint  $x^2 + y^2 + z^2 < 225$  in order that the problem have a solution, albeit one very far removed from the "false solution" to which the simpler algorithm converged. By  $k = 5$  the third coordinate had exceeded  $\frac{4+\sqrt{2}}{4}$ , by  $k = 82$  the artificial constraint was binding, and the algorithm successfully terminated after 111 iterations.

Example 6. Finally the algorithm was tested on a family of problems each involving two variables ( $d = 2$ ), three  $v_j$ s ( $n = 3$ ), and four constraints ( $m = 4$ ). All functions were quadratic of the form  $a_1(x-a_2)^2 + a_3(y-a_4)^2 - a_5$ , and various trials were conducted with different choices of the parameters  $a_i$ . For example, consider the two problems with  $v_1(x,y) = (x-2)^2 + y^2$ ,  $v_2(x,y) = (1/2)(x^2 + y^2)$ ,  $v_3(x,y) = x^2 + (y-2)^2$ ,  $g_1(x,y) = (x-1)^2 + y^2 - 4$ ,  $g_2(x,y) = (x+1)^2 + y^2 - 4$ ,  $g_3(x,y) = x^2 + (y-1)^2 - 4$ ,  $g_4(x,y) = x^2 + (y+1)^2 - 4$ . With  $f(x,y) = (x-2)^2 + (y-2)^2$  the solution  $(a,a)$ ,  $a = \frac{\sqrt{7}-1}{2}$ , was found in 1 step, while with  $f(x,y) = (x-4)^2 + (y-1)^2$  the same optimum point was found in 9 iterations. In each case  $X_0 = (0,0)$  was obtained by the fully implemented algorithm and  $\epsilon_0 = 1.5$ .

## 7. Conclusions

As Rubin [11] found with the earlier, linear version of this algorithm, the subgradient projection algorithm is a viable method for solving a number of convex programming problems including some which do not meet the hypotheses needed to guarantee convergence, e.g. examples 1,2,4. When applied to smooth, ie. differentiable, problems, e.g. example 3, it appears to perform competitively with less general algorithms designed specifically to solve such problems. It successfully avoids jamming on test problems that have been specifically designed to jam, e.g. example 4. It qualitatively replicates the results of the earlier linear method when applied to previously tested problems. And it successfully solves in a small number of iterations larger test problems designed to exercise the full range of capabilities built into the algorithm.

Since one always wants faster convergence than is available, it might be worth speculating on what factors might limit the performance of this algorithm. The anti-jamming techniques, although clearly necessary, seem to entail a rather heavy cost in slowing down performance. Whereas the gradient and even the subdifferential are local entities, the  $\epsilon$ -subdifferential is not. While many  $\epsilon$ -subgradient based algorithms require knowledge of the entire  $\epsilon$ -subdifferential, a formidable if not prohibitive task, Sreedharan's algorithm requires only an easily computable subset of that set. An appropriate element of the  $\epsilon$ -subdifferential is chosen and used to construct a feasible direction of strict descent while avoiding jamming. One possible source of difficulty is the difference between the directional derivative  $F'(x;u)$  and the  $\epsilon$ -subgradient based approximation to it, say  $F'_\epsilon(x;u)$ , where  $F$  is the objective function, e.g.  $F = f + v$  in our case. Hiriart-Urruty [6] has found that  $|F'(x;u) - F'_\epsilon(x;u)| = O(\sqrt{\epsilon})$  as  $\epsilon \rightarrow 0$ . In some of

our trials, indirect evidence of this qualitative behavior can be found indicating a possible culprit when slow convergence is observed. Consider the following simple problem:

$$\text{Minimize: } F(x,y) = (x-3)^2 + (y-1)^2$$

$$\text{Subject to: } x^4 + y^4 < 1$$

Referring to Table III, the correspondence either between  $\epsilon_k$  and the number of correct figures of the iterate or between  $\epsilon_k$  and  $\|y_0\|_\infty$  suggests an  $\sqrt{\epsilon}$ -order convergence rate in line with Hiriart-Urruty's result.

$k$	$(x_k, y_k)$	$\epsilon_k$	$\ y_0\ _\infty$	$F(x_k, y_k)$
0	(0, 0)	.5	6	10
5	(.970, .58)	$5 \times 10^{-4}$	$2.67 \times 10^{-2}$	4.29559
10	(.97122, .5762395)	$5 \times 10^{-10}$	$4.38 \times 10^{-5}$	4.29553627388
15	(.97121490, .5762462)	$5 \times 10^{-13}$	$1.08 \times 10^{-6}$	4.2955362736910
20	(.97121493596, .5762460176)	$5 \times 10^{-16}$	$1.13 \times 10^{-9}$	4.295536273690917
*	(.9712149358190, .5762460177155)			

Table III

Indirect or suggestive evidence of a  $\sqrt{\epsilon}$ -order rate of convergence of the subgradient projection algorithm

\* The actual solution correct to 12 digits.

## REFERENCES

1. M. Avriel, "Nonlinear Programming: Analysis and Methods", Prentice-Hall, Inc., Englewood Cliffs, N.J., 1976.
2. A. Ben-Israel, A. Ben-Tal, and S. Zlobec, "Optimality in Nonlinear Programming: A Feasible Directions Approach", John Wiley & Sons, New York, 1981.
3. D. P. Bertsekas and S. K. Mitter, A descent numerical method for optimization problems with nondifferentiable cost functionals, SIAM J. Control 11 (1973), 4, 637-652.
4. R. W. Cottle, The principal pivoting method of quadratic programming, Mathematics of the Decision Sciences, 1 (1968), 144-162.
5. V. F. Demyanov and V. N. Malozemov, The theory of nonlinear minimax problems, Uspekhi Mat. Nauk., 26 (1971), 53-104.
6. J.-B. Hiriart-Urruty,  $\epsilon$ -subdifferential calculus, in "Convex Analysis and Optimization", ed. by J. P. Aubin and R. B. Vinter, Pitman, Research Notes in Math., 57, London, 1982.
7. E. Polak, "Computational Methods in Optimization: A Unified Approach", Academic Press, New York, 1971.
8. QUADMP/QUADPR, Academic Computing Center, University of Wisconsin-Madison, 1981.
9. J. B. Rosen, The gradient projection method for nonlinear programming, Part I: Linear constraints, J. SIAM 8 (1960), 181-217.
10. J. B. Rosen, The gradient projection method for nonlinear programming, Part II: Nonlinear constraints, J. SIAM 9 (1961), 514-532.
11. P. A. Rubin, Implementation of a subgradient projection algorithm, Inter. J. Computer Maths. 12 (1983), 321-328.
12. V. P. Sreedharan, A subgradient projection algorithm, J. Approx. Theory 35 (1982), 2, 111-126.
13. V. P. Sreedharan, Another subgradient projection algorithm, to appear.
14. P. Wolfe, A method of conjugate subgradients for minimizing nondifferentiable functions, Math. Prog. Study 3 (1975), 145-173.

RWO/jik

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER  2550	2. GOVT ACCESSION NO. <b>ADA132 835</b>	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle)  IMPLEMENTATION OF A SUBGRADIENT PROJECTION ALGORITHM II		5. TYPE OF REPORT & PERIOD COVERED Summary Report - no specific reporting period
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s)  Robert W. Owens		8. CONTRACT OR GRANT NUMBER(s)  DAAG29-80-C-0041
9. PERFORMING ORGANIZATION NAME AND ADDRESS Mathematics Research Center, University of 610 Walnut Street Madison, Wisconsin 53706		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Work Unit Number 5 - Optimization and Large Scale Systems
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Research Office P.O. Box 12211 Research Triangle Park, North Carolina 27709		12. REPORT DATE August 1983
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		13. NUMBER OF PAGES 18
		15. SECURITY CLASS. (of this report)  UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)  Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)  optimization, subgradient, convex function, algorithm		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)  This paper discusses the implementation of a subgradient projection algorithm due to Sreedharan [13] for the minimization, subject to a finite number of smooth, convex constraints, of an objective function which is the sum of a smooth, strictly convex function and a piecewise smooth convex function. Computational experience with the algorithm on several test problems and comparison of this experience with previously published results is presented.		



ND

DATE

LMED

83

OTIC